

040 Lazy Streams [QUIZ, 10 minutes]

Thank you for working alone. Help others by remaining quiet after you have finished.

Exercise 1. Write a function `even_from` that generates an infinite stream of even numbers starting from (a possibly odd) number `n`, its only parameter. You can use any functions from the book or from the exercise.

Exercise 2. Implement `flatten`: `Stream[Stream[A]] => Stream[A]`. This function preserves the contents of the input stream of streams, but simplifies its structure into a single stream. You can use any functions defined in the book or exercises.

An example solution for question 1:

```
def even_from (n:Int ) =from (n).filter (_%2==0)
```

A half point for an eager solution (a lazy solution should return without evaluating the tail of the stream). One point for a lazy solution with some flaws. Two points for a solution with only minor flaws.

An example solution for question 2:

```
def flatten[A] (ss: Stream[Stream[A]]) :Stream[A] =ss.flatMap (s =>s)
```

A half point for an eager solution (a lazy solution should return without evaluating the tail of the stream). One point for a lazy solution with some flaws. Two points for a solution with only minor flaws.

PASS: 3-4 points